



Conservative fourth-order time integration of non-linear dynamic systems

Krenk, Steen

Published in:
Computer Methods in Applied Mechanics and Engineering

Link to article, DOI:
[10.1016/j.cma.2015.06.016](https://doi.org/10.1016/j.cma.2015.06.016)

Publication date:
2015

Document Version
Peer reviewed version

[Link back to DTU Orbit](#)

Citation (APA):
Krenk, S. (2015). Conservative fourth-order time integration of non-linear dynamic systems. *Computer Methods in Applied Mechanics and Engineering*, 295, 39-55. <https://doi.org/10.1016/j.cma.2015.06.016>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Conservative Fourth-order Time Integration of Non-linear Dynamic Systems

Steen Krenk

*Department of Mechanical Engineering,
Technical University of Denmark, DK-2800 Lyngby, Denmark*

Abstract

An energy conserving time integration algorithm with fourth-order accuracy is developed for dynamic systems with nonlinear stiffness. The discrete formulation is derived by integrating the differential state-space equations of motion over the integration time increment, and then evaluating the resulting time integrals of the inertia and stiffness terms via integration by parts. This process introduces the time derivatives of the state space variables, and these are then substituted from the original state-space differential equations. The resulting discrete form of the state-space equations is a direct fourth-order accurate representation of the original differential equations. This fourth-order form is energy conserving for systems with force potential in the form of a quartic polynomial in the displacement components. Energy conservation for a force potential of general form is obtained by addition of a higher order secant-type correction term. The formulation leads to a consistent representation of the motion within a time increment corresponding to cubic Hermite interpolation in time. This in turn leads to excellent phase representation with only a small fourth-order error, permitting integration of oscillatory systems with only a few integration points per period. Three numerical examples demonstrate the high accuracy of the algorithm.

Keywords: Nonlinear dynamics, Time integration, Energy conservation, Fourth-order accuracy.

Email address: `sk@mek.dtu.dk` (Steen Krenk)

1. Introduction

The numerical calculation of the response of dynamic systems is one of the central problems in computational mechanics and physics. In the last decades there has been a considerable interest in the development of time integration methods that conserve invariants of the problem like energy and momentum. An early contribution is the series of papers by LaBudde and Greenspan [1, 2], who developed discrete time integration methods for particle dynamics based on energy and momentum conservation. This type of approach to the numerical calculation of nonlinear dynamic response was further developed by Simo et al. [3, 4, 5] covering rigid-body motion, elastodynamics and general Hamiltonian systems. An essential point in these methods is the identification of a particular representation of the internal forces such that exact conservation properties are satisfied. In the context of continuum mechanics various forms of stress averages have recently been discussed by Romero [6].

An important extension of the range of conservative time integration methods was attained by the introduction of the so-called finite derivative introduced by Gonzalez [7, 8], and further developed under the name of the ‘discrete gradient’ by McLachlan et al. [9]. The basic idea is to replace the original expression of the internal force by an augmented form including a correction in terms of the increment of the quantity to be conserved. This concept bears similarity with the notion of a secant representation of derivatives used in quasi Newton methods, see e.g. [10] Chapter 9. The concept was rapidly adopted within computational mechanics, see e.g. [11, 12, 13], and the finite derivative is now considered a standard concept in time integration of nonlinear response, [14]. Extensions have been made to the original second-order representation, e.g. by improved evaluation of the integral via Hermite type integration with the velocities as the derivatives at the time integration points [15]. A systematic extension of second-order algorithms to fourth-order accuracy has been proposed independently by de Frutos and Sanz-Serna [16], and Tarnow and Simo [17]. The idea is to combine three steps of the second-order algorithm to cancel the leading error terms and increasing the accuracy to fourth order for non-dissipative problems, where time can be reversed. By this approach the fourth-order algorithm inherits properties like energy conservation

from the underlying second-order algorithm.

Most direct time integration schemes with conservation properties are of second-order accuracy with respect to the representation of the response. Essentially, the conservative algorithms correspond to an integrated form of the original state-space differential equations, and the order of accuracy depends on the order to which the corresponding integrals are represented in the discrete algorithm. It was demonstrated by Krenk [18] that a fourth-order algorithm could be obtained for linear dynamic systems by evaluating the integrals over the time step via integration by parts. This procedure introduces the time derivatives of the state-space variables under the integral sign, and these time derivatives can then be expressed via the original state-space equations. The resulting integrals contain time-weighted forms of the state-space variables and these can be evaluated to fourth order by a Taylor series expansion about the center of the time step. The time-weighted integrals bear a certain similarity to the format of Galerkin based time integration methods, see e.g. the higher order Galerkin based schemes of Gross et al. [19], and it has recently been demonstrated by Depouhon et al. [20] that within a linear framework a discontinuous Galerkin procedure presented by Bottasso and Trainelli [21] can be reduced by elimination of the discontinuities to the direct state-space format from [18]. In the present paper the direct fourth-order state-space format is extended to nonlinear systems, and the terms of the algorithm are expressed entirely by the forces and the stiffness matrix at the end points of each time step. This results in a fourth-order accurate time integration algorithm that is energy conserving, if the energy potential is of fourth degree or less in the displacement variables. The algorithm is easily extended to energy conserving form also for more general internal energy potentials by introducing a secant type correction term of fifth order, thereby retaining the fourth order accuracy of the response history. Examples demonstrate very high accuracy of the algorithm, and a closer look at the discrete equations of the algorithm reveals that it incorporates a full cubic Hermite representation of the response within each integration step, whereas similar second order algorithms typically are based on a central difference representation of the mean velocity, corresponding to only a quadratic representation.

2. Basic equations

A standard form of the equations of motion of dynamic systems with non-linear internal forces described by a finite number of degrees of freedom \mathbf{u} is

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{g}(\mathbf{u}) = \mathbf{f}(t). \quad (1)$$

A dot denotes time differentiation, while \mathbf{M} and \mathbf{C} are the mass matrix and damping matrix, respectively, here assumed to be constant. Alternative, more general forms, in which \mathbf{u} contains the generalized coordinates and the inertial terms depend on both the generalized displacements and the generalized velocities, can be formulated by use of Hamilton's principle or by the Lagrange equations with the momentum vector \mathbf{p} as an auxiliary variable, see e.g. [22]. However, the present form has wide applications e.g. for simple nonlinear mechanical systems, for particle dynamics, and for continuum mechanics modeled by isoparametric elements. The internal force $\mathbf{g}(\mathbf{u})$ is assumed to correspond to an energy potential $G(\mathbf{u})$, such that $\mathbf{g} = \nabla_{\mathbf{u}}G(\mathbf{u})$, where $\nabla_{\mathbf{u}}$ denotes the gradient with respect to the components of \mathbf{u} . The external load vector \mathbf{f} , conjugate to the displacement vector \mathbf{u} , is typically given either directly as a function of time as $\mathbf{f}(t)$ or as a potential force of the form $\mathbf{f} = -\nabla_{\mathbf{u}}F(\mathbf{u})$. In the present derivation of the algorithm only the direct time-dependent part of the load $\mathbf{f}(t)$ is included explicitly, as any potential load can be included by replacing the internal energy potential $G(\mathbf{u})$ by the combined potential $G(\mathbf{u}) + F(\mathbf{u})$ in the final formulation.

The present formulation makes extensive use of the relation between derivatives and increments over the integration interval. In particular this introduces the tangent stiffness matrix, defined by

$$\mathbf{K}(\mathbf{u}) = \nabla_{\mathbf{u}}^T \mathbf{g}(\mathbf{u}) = \nabla_{\mathbf{u}}^T \nabla_{\mathbf{u}} G(\mathbf{u}), \quad (2)$$

into the discrete dynamic equations. This is in contrast to traditional formulations, in which the stiffness matrix is only used for iterative solution of a balance equation containing the forces defined in the continuous problem.

2.1. State-space equations

The goal of a single-step time integration algorithm is to advance the state-space variables $\mathbf{u}, \dot{\mathbf{u}}$ from a time t_n to the time $t_{n+1} = t_n + h$, where h is the integration

time step, by use of the system matrices and the history of the load vector in the time interval. In this process it is advantageous to work with a state-space formulation, in which the velocity is introduced as a new independent variable $\mathbf{v} = \dot{\mathbf{u}}$. When this kinematic relation is multiplied by the mass matrix \mathbf{M} a system of first-order state-space equations is obtained in the following symmetric format,

$$\begin{bmatrix} \mathbf{C} & \mathbf{M} \\ \mathbf{M} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{v}} \end{bmatrix} + \begin{bmatrix} \mathbf{g}(\mathbf{u}) \\ -\mathbf{M}\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{f}(t) \\ \mathbf{0} \end{bmatrix}. \quad (3)$$

There are two advantages of using the state-space vector $[\mathbf{u}^T, \mathbf{v}^T]$ when deriving a discrete time integration scheme: the definition of $\mathbf{v}(t)$ as being identical to the velocity $\dot{\mathbf{u}}(t)$ at all times can be relaxed and included in the approximate procedure, and symmetry of the equations can be retained and related to energy conservation in a simple manner. In this context it is important that the viscous damping is included in the form $\mathbf{C}\dot{\mathbf{u}}$, and therefore appears as a symmetric matrix on the diagonal of the first block matrix.

2.2. Energy balance equation

An energy balance relation of the system can be obtained directly from the state-space equations of motion (3) by pre-multiplication with $[\dot{\mathbf{u}}^T, -\dot{\mathbf{v}}^T]$. In this process the contributions from the two off-diagonal sub-matrices of the first term cancel, and when using the ‘chain rule’ $\dot{G}(\mathbf{u}) = \dot{\mathbf{u}}^T \mathbf{g}(\mathbf{u})$ the following energy balance equation is obtained,

$$\frac{d}{dt} \left[\frac{1}{2} \mathbf{v}^T \mathbf{M} \mathbf{v} + G(\mathbf{u}) \right] = \dot{\mathbf{u}}^T \mathbf{f} - \dot{\mathbf{u}}^T \mathbf{C} \dot{\mathbf{u}}. \quad (4)$$

The left side is the time derivative of the total system energy, while the terms on the right side are the rate of work of the external force and the viscous dissipation, respectively. It is desirable for a numerical time integration algorithm to have a similar energy balance equation in finite increment form. In addition to serving as an important characteristic of a conservative system, the energy also serves as a norm by which energy conservation guarantees stability of the corresponding time integration scheme. If it is desired to introduce algorithmic damping in addition to the damping represented by the system

damping matrix \mathbf{C} in order to attenuate potentially under-sampled high-frequency components, the ideal is to have it in the form of an additional negative quadratic form on the right side. However, this aspect will not be pursued in the present paper.

3. Discrete momentum formulation

The first step in the derivation of a momentum based finite step time integration algorithm is to integrate the state-space equations of motion (3) over the integration time interval $[t_n, t_{n+1}]$ of length $h = t_{n+1} - t_n$. The first term containing the derivatives $\dot{\mathbf{u}}(t)$ and $\dot{\mathbf{v}}(t)$ can be integrated exactly, giving the finite increments $\Delta\mathbf{u}$ and $\Delta\mathbf{v}$ over the time interval, whereby the integrated state-space equations take the form

$$\begin{bmatrix} \mathbf{C} & \mathbf{M} \\ \mathbf{M} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{v} \end{bmatrix} + \begin{bmatrix} \int \mathbf{g}(\mathbf{u}) dt \\ - \int \mathbf{M}\mathbf{v} dt \end{bmatrix} = \begin{bmatrix} \int \mathbf{f} dt \\ \mathbf{0} \end{bmatrix}. \quad (5)$$

The first of these equations expresses momentum balance via the corresponding increments over the time interval, while the second equation defines the kinematic relation between the displacement increment $\Delta\mathbf{u}$ and the velocity \mathbf{v} within the integration interval.

The key to the numerical solution of these equations lies in the representation of the three time integrals. In order to obtain a fourth-order accurate method the integrals must be discretized by suitable fourth-order representations. If the energy function $G(\mathbf{u})$ is of polynomial form of fourth degree or less, fourth-order integration will imply energy conservation, while for a general energy function energy conservation can be obtained by including an additional secant-based term in the algorithm. The following subsections develop suitable fourth-order representations of the time integrals of the external force, the momentum, and the internal force. The additional condition of energy conservation is then discussed subsequently in Section 4.

3.1. External force integral

In the present formulation $\mathbf{f}(t)$ represents the part of the load given explicitly as a function of time. In general the load may also have contributions from an external force potential $F(\mathbf{u})$, and these contributions can then be represented by replacing the internal

force potential $G(\mathbf{u})$ by the combined potential $G(\mathbf{u}) + F(\mathbf{u})$ as mentioned above. The time integral of the external force $\mathbf{f}(t)$ is expressed in terms of the algorithmic external force

$$\mathbf{f}_* = \frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{f}(t) dt. \quad (6)$$

In order to obtain a fourth-order algorithm the integral must be represented to fourth order. For simple load histories, such as harmonic load, the integral can be evaluated directly in analytical form. Alternatively, approximate fourth-order approximations can be obtained from a suitable Taylor expansion of $\mathbf{f}(t)$, in particular by including a mid-point value or by using the time derivative at the interval end-points.

Let the function $\mathbf{f}(t)$ be represented within the time interval $[t, t+h]$ by its Taylor expansion around the interval center $t_{n+1/2}$ in terms of the time variable $\tau = t - t_{n+1/2}$. The time integral then takes the form

$$\mathbf{f}_* = \frac{1}{h} \int_{t_n}^{t_{n+1}} (\mathbf{f}_{n+1/2} + \frac{1}{2} \ddot{\mathbf{f}}_{n+1/2} \tau^2 + \dots) dt = \mathbf{f}_{n+1/2} + \frac{1}{24} h^2 \ddot{\mathbf{f}}_{n+1/2} + O(h^4), \quad (7)$$

where the dot symbol is used for time differentiation. The Taylor expansion of the interval end-point mean value is

$$\bar{\mathbf{f}} = \frac{1}{2}(\mathbf{f}_{n+1} + \mathbf{f}_n) = \mathbf{f}_{n+1/2} + \frac{1}{8} h^2 \ddot{\mathbf{f}}_{n+1/2} + O(h^4) \quad (8)$$

The second derivative $\ddot{\mathbf{f}}_{n+1/2}$ is now eliminated from the integral mean, whereby

$$\mathbf{f}_* = \frac{1}{3} \bar{\mathbf{f}} + \frac{2}{3} \mathbf{f}_{n+1/2} + O(h^4). \quad (9)$$

This formula corresponds to Simpson integration, where the introduction of a suitable average of the end-point values and the center value $\mathbf{f}_{n+1/2}$ increases the order of the representation from second to fourth order.

As an alternative to the internal point representation (9), a representation can be obtained in terms of the end-point mean value and the increment of the time derivative $\dot{\mathbf{f}}$. First, the mid-point value $\mathbf{f}_{n+1/2}$ is eliminated between (7) and (8), whereby

$$\mathbf{f}_* = \bar{\mathbf{f}} - \frac{1}{12} h^2 \ddot{\mathbf{f}}_{n+1/2} + O(h^4). \quad (10)$$

By using the Taylor expansion

$$\Delta \dot{\mathbf{f}} = \dot{\mathbf{f}}_{n+1} - \dot{\mathbf{f}}_n = h \ddot{\mathbf{f}}_{n+1/2} + O(h^3), \quad (11)$$

the double derivative $\ddot{\mathbf{f}}_{n+1/2}$ can be eliminated from (10), whereby the representation takes the form

$$\mathbf{f}_* = \bar{\mathbf{f}} - \frac{1}{12}h\Delta\dot{\mathbf{f}} + O(h^4). \quad (12)$$

In this Hermitian form the dependence on an internal value $\mathbf{f}_{n+1/2}$ has been replaced by dependence on the increment $\Delta\dot{\mathbf{f}}$ of the time derivative over the interval. While this tradeoff may not be particularly advantageous when representing the load time history, it represents a key to formulating an efficient representation of the internal force solely in terms of end-point values as discussed later.

In conclusion, the part of the load given as a function of time may be introduced into the algorithm either in the form of the time integral (6), as an arithmetic mean (9) including the interval center value, or in the Hermitian form (12) including the load and the increment of its time derivative over the interval.

3.2. Momentum integral

Following [18], the momentum time integral is evaluated by using the interval centered time variable $\tau = t - t_{n+1/2}$ in connection with integration by parts,

$$\int_{t_n}^{t_{n+1}} \mathbf{M}\mathbf{v} dt = [\mathbf{M}\mathbf{v}\tau]_n^{n+1} - \int_{t_n}^{t_{n+1}} \mathbf{M}\dot{\mathbf{v}}\tau dt. \quad (13)$$

In the integral on the right side the derivative of the velocity $\dot{\mathbf{v}}$ is substituted from the first of the state-space equations (2), whereby the momentum integral takes the form

$$\int_{t_n}^{t_{n+1}} \mathbf{M}\mathbf{v} dt = h\mathbf{M}\bar{\mathbf{v}} + \int_{t_n}^{t_{n+1}} (\mathbf{C}\dot{\mathbf{u}} + \mathbf{g}(\mathbf{u}) - \mathbf{f})\tau dt. \quad (14)$$

In the last integral the substitution $\mathbf{v} = \dot{\mathbf{u}}$ is introduced in the damping term. The integrand then is replaced by its Taylor expansion around the interval center time $t_{n+1/2}$, whereby

$$\int_{t_n}^{t_{n+1}} (\mathbf{C}\mathbf{v} + \mathbf{g}(\mathbf{u}) - \mathbf{f})\tau dt = \int_{t_n}^{t_{n+1}} (\mathbf{C}\dot{\mathbf{v}}_{n+1/2} + \mathbf{K}_{n+1/2}\dot{\mathbf{u}}_{n+1/2} - \dot{\mathbf{f}}_{n+1/2})\tau^2 dt + O(h^5). \quad (15)$$

In this integral the first derivatives are expressed in terms of the corresponding function increment over the time interval $[t_n, t_{n+1}]$ according to the Taylor expansion

$$\Delta\mathbf{f} = \mathbf{f}_{n+1} - \mathbf{f}_n = h\dot{\mathbf{f}}_{n+1/2} + O(h^3), \quad (16)$$

and similarly for $\dot{\mathbf{u}}_{n+1/2}$ and $\dot{\mathbf{v}}_{n+1/2}$. By using these representations in (15) the momentum integral (14) takes the form

$$\int_{t_n}^{t_{n+1}} \mathbf{M}\mathbf{v} dt = h\mathbf{M}\bar{\mathbf{v}} + \frac{1}{12}h^2(\mathbf{C}\Delta\mathbf{v} + \mathbf{K}_{n+1/2}\Delta\mathbf{u} - \Delta\mathbf{f}) + O(h^5). \quad (17)$$

The stiffness matrix $\mathbf{K}_{n+1/2}$ corresponds to the center time $t_{n+1/2}$. However, it follows from Taylor expansion around $t_{n+1/2}$ that

$$\bar{\mathbf{K}} = \frac{1}{2}(\mathbf{K}_{n+1} + \mathbf{K}_n) = \mathbf{K}_{n+1/2} + O(h^2). \quad (18)$$

Thus, the momentum integral (17) retains its order of accuracy, if the stiffness matrix $\mathbf{K}_{n+1/2}$ at the center time $t_{n+1/2}$ is replaced by the computationally more convenient mean of its end-point values $\bar{\mathbf{K}}$, whereby

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{M}\mathbf{v} dt = \mathbf{M}\bar{\mathbf{v}} + \frac{1}{12}h(\mathbf{C}\Delta\mathbf{v} + \bar{\mathbf{K}}\Delta\mathbf{u} - \Delta\mathbf{f}) + O(h^4). \quad (19)$$

The second term on the right, involving the parenthesis, is of second order and is typically not included in second-order algorithms.

3.3. Internal force time integral

Two fourth-order representations of the time mean of the internal force are obtained by use of the results for the external force integral in section 3.1. The first of these representations makes use of an internal point, while the second is of Hermitian type and uses the internal force and the stiffness matrix at the interval end-points.

Internal point formula

The internal point formula for the time mean value of the internal force follows from (9), which is essentially Simpson's formula,

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \frac{1}{3}\bar{\mathbf{g}} + \frac{2}{3}\mathbf{g}(\mathbf{u}_{n+1/2}) + O(h^4). \quad (20)$$

In this formula the center value is evaluated at the displacement at the center of the time interval, $\mathbf{u}_{n+1/2} = \mathbf{u}(t_{n+1/2})$. The relation between this value and the algebraic mean follows from Taylor expansion in the same way as (8) for the force integral,

$$\bar{\mathbf{u}} = \frac{1}{2}(\mathbf{u}_{n+1} + \mathbf{u}_n) = \mathbf{u}_{n+1/2} + \frac{1}{8}h^2\ddot{\mathbf{u}}_{n+1/2} + O(h^4). \quad (21)$$

The acceleration $\ddot{\mathbf{u}}_{n+1/2}$ at the time mid-point is now expressed by Taylor expansion in terms of the velocity increment,

$$\Delta \mathbf{v} = \Delta \dot{\mathbf{u}} = h \ddot{\mathbf{u}}_{n+1/2} + O(h^3). \quad (22)$$

Elimination of the center point acceleration $\ddot{\mathbf{u}}_{n+1/2}$ between (21) and (22) then gives the following expression for the displacement at the center time $t_{n+1/2}$,

$$\mathbf{u}_{n+1/2} = \bar{\mathbf{u}} - \frac{1}{8}h\Delta \mathbf{v} + O(h^4). \quad (23)$$

It is seen that the difference between the algebraic mean value $\bar{\mathbf{u}}$ and the center-time value $\mathbf{u}_{n+1/2}$ is given by the second order term $\frac{1}{8}h\Delta \mathbf{v}$, involving the increment of the velocity. This term, accounting for the fact that the displacement may follow a curved path within the time increment, is closely associated with the use of the time-integrated form of the equations of motion and is typically not included in second order algorithms.

The center-time value of the internal force $\mathbf{g}_{n+1/2}$ is now expressed in terms of mean values by the Taylor expansion

$$\mathbf{g}(\mathbf{u}_{n+1/2}) = \mathbf{g}(\bar{\mathbf{u}}) + \mathbf{K}(\bar{\mathbf{u}})(\mathbf{u}_{n+1/2} - \bar{\mathbf{u}}) + O(h^4). \quad (24)$$

In this formula the stiffness matrix at the mean displacement can be replaced by the mean value of the stiffness matrix to within second order as shown in (18), and the displacement difference can similarly be expressed to second order by (23), resulting in the fourth-order relation

$$\mathbf{g}(\mathbf{u}_{n+1/2}) = \mathbf{g}(\bar{\mathbf{u}}) - \frac{1}{8}h\bar{\mathbf{K}}\Delta \mathbf{v} + O(h^4). \quad (25)$$

Substitution of this result into the time mean value formula (20) gives the following internal point formula in terms of the displacement mean,

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \frac{1}{3}\bar{\mathbf{g}} + \frac{2}{3}\mathbf{g}(\bar{\mathbf{u}}) - \frac{1}{12}h\bar{\mathbf{K}}\Delta \mathbf{v} + O(h^4). \quad (26)$$

It is seen that two factors contribute to the fourth-order accuracy of the integral representation in the case of non-linear internal forces: The internal force is evaluated both at the interval end points and at the mean displacement, and the curvature of the displacement path introduces an additional term involving the velocity increment and the mean value of the stiffness matrix.

Hermitian internal force formula

An alternative formula for the time-mean value of the internal force can be obtained by application of the Hermitian integration formula (12) to the internal force $\mathbf{g}(\mathbf{u})$,

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \bar{\mathbf{g}} - \frac{1}{12} h \Delta(\mathbf{K} \dot{\mathbf{u}}) + O(h^4). \quad (27)$$

In this formula it is used that a finite difference of a product is equal to the sum of the difference of each factor multiplied by the mean value of the other factor. This reduces the integration formula to

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \bar{\mathbf{g}} - \frac{1}{12} h (\Delta \mathbf{K} \bar{\mathbf{u}} + \bar{\mathbf{K}} \Delta \dot{\mathbf{u}}) + O(h^4). \quad (28)$$

The final form is obtained by introducing the second-order approximation $\Delta \mathbf{u} \simeq h \bar{\mathbf{u}}$ and the identity $\Delta \dot{\mathbf{u}} = \Delta \mathbf{v}$, whereby

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \bar{\mathbf{g}} - \frac{1}{12} \Delta \mathbf{K} \Delta \mathbf{u} - \frac{1}{12} h \bar{\mathbf{K}} \Delta \mathbf{v} + O(h^4). \quad (29)$$

The two incremental terms are of second order and are both necessary for a fourth-order representation of the time-mean value of the internal force. It is noted that the present formula avoids the need for evaluating the internal force at the mean displacement $\bar{\mathbf{u}}$.

4. Energy conservation

A discrete form of the state-space equations (5) is obtained by substitution of the fourth-order time integral representations of the previous section. Hereby the state-space equations take the block-matrix form

$$\begin{bmatrix} \mathbf{C} & \mathbf{M} - \frac{1}{12} h^2 \bar{\mathbf{K}} \\ \mathbf{M} - \frac{1}{12} h^2 \bar{\mathbf{K}} & -\frac{1}{12} h^2 \mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u} \\ \Delta \mathbf{v} \end{bmatrix} + h \begin{bmatrix} \mathbf{g}_*(\mathbf{u}) \\ -\mathbf{M} \bar{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} h \mathbf{f}_* \\ -\frac{1}{12} h^2 \Delta \mathbf{f} \end{bmatrix}. \quad (30)$$

The time-dependent part of the external force, denoted by \mathbf{f}_* in the algorithm, can be either the actual time integral (6) or any fourth-order representation such as (9) or (12).

In the equations (30) the velocity term in the time integral mean values (26) and (29) of the internal force has been extracted and represented explicitly in the first block-matrix, which then becomes symmetric. This leaves the algorithmic internal force \mathbf{g}_* as

defined to within fourth order by the time integral mean value in the form

$$\frac{1}{h} \int_{t_n}^{t_{n+1}} \mathbf{g}(\mathbf{u}) dt = \mathbf{g}_* - \frac{1}{12} h \bar{\mathbf{K}} \Delta \mathbf{v} + O(h^4). \quad (31)$$

Two fourth-order representations of the internal force have been obtained in the previous section, and thus the algorithmic force has the form

$$\mathbf{g}_* = \mathbf{g}_q + O(h^4). \quad (32)$$

where \mathbf{g}_q is a fourth-order representation, e.g. from (26) or (29), or a linear combination of these,

$$\mathbf{g}_q = \begin{cases} \frac{1}{3} \bar{\mathbf{g}} + \frac{2}{3} \mathbf{g}(\bar{\mathbf{u}}), \\ \bar{\mathbf{g}} - \frac{1}{12} \Delta \mathbf{K} \Delta \mathbf{u}, \\ \mathbf{g}(\bar{\mathbf{u}}) + \frac{1}{24} \Delta \mathbf{K} \Delta \mathbf{u}. \end{cases} \quad (33)$$

These three fourth-order representations correspond to modified forms of the Simpson rule, the trapezoidal rule, and a center-point rule, respectively. In the case where the energy potential $G(\mathbf{u})$ is a fourth degree polynomial in the displacement components, the higher order terms $O(h^4)$ in (31) and (32) vanish, and thus in this case all three expressions are identical and \mathbf{g}_q is the algorithmic force. In fact, in the case of linear elasticity with quadratic strains the potential is quartic and the algorithmic force \mathbf{g}_q can also be calculated at an internal point combining the mean strain and the mean stress [4]. The various forms are derived by simple means and illustrated in [23]. For a general internal energy potential, energy conservation is obtained by an extended definition of the algorithmic force \mathbf{g}_* , discussed below.

The energy balance equation of the discrete state-space equations (30) is obtained by pre-multiplication with $[\Delta \mathbf{u}^T, -\Delta \mathbf{v}^T]$, leading to

$$\begin{aligned} \left[\frac{1}{2} \mathbf{v}^T \mathbf{M} \mathbf{v} \right]_n^{n+1} + \Delta \mathbf{u}^T \mathbf{g}_*(\mathbf{u}) &= \Delta \mathbf{u}^T \mathbf{f}_* + \frac{1}{12} h \Delta \mathbf{v}^T \Delta \mathbf{f} \\ &\quad - \frac{1}{h} (\Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u} + \frac{1}{12} h^2 \Delta \mathbf{v}^T \mathbf{C} \Delta \mathbf{v}) \end{aligned} \quad (34)$$

This equation is the discrete algorithmic representation of the energy balance equation (4). In order to represent proper energy balance the second term on the left must equal

the increment of the internal energy, corresponding to the following condition on the algorithmic internal force,

$$\Delta G = \Delta \mathbf{u}^T \mathbf{g}_* . \quad (35)$$

The combination of asymptotic match with the gradient implied by (32) and higher-order terms leading to satisfaction of the energy increment condition (35) defines \mathbf{g}_* as a finite derivative of the internal energy $G(\mathbf{u})$. In the case of a quartic potential function $G(\mathbf{u})$ the algorithmic internal force is given by any of the expressions in (33). In the general case the use of this form as an approximation would lead to an error in the energy increment of the form

$$\Delta G - \Delta \mathbf{u}^T \mathbf{g}_q = O(\|\Delta \mathbf{u}\|^5) = O(h^5) . \quad (36)$$

Thus, for a general potential function $G(\mathbf{u})$ the representation of the internal force integral in the discrete time integration algorithm would lead to an error of order $O(h^5)$ in the energy balance.

Exact energy conservation for free undamped response can be established by including the energy difference (36) in the form of a correction term in secant format. A second-order secant approximation was introduced in the form of a finite derivative or gradient into computational mechanics by Gonzalez [7, 8] and McLachlan et al. [9]. The finite derivative is a special form of a secant correction, used to obtain a special property over a finite interval, see e.g. [10]. In the present case the secant representation is used to provide a higher order correction to the fourth-order representation \mathbf{g}_q in the form

$$\mathbf{g}_* = \mathbf{g}_q + \frac{\Delta G - \Delta \mathbf{u}^T \mathbf{g}_q}{\Delta \mathbf{u}^T \bar{\mathbf{K}} \Delta \mathbf{u}} \bar{\mathbf{K}} \Delta \mathbf{u} . \quad (37)$$

where $\bar{\mathbf{K}}$ is the mean stiffness matrix, serving to define a consistent second-order norm of the displacement increment vector $\Delta \mathbf{u}$. In the case of a quartic potential $G(\mathbf{u})$, the additional secant correction term in (37) vanishes, leaving the direct representation $\mathbf{g}_* = \mathbf{g}_q$. The exact representation of the energy increment follows directly from pre-multiplication of \mathbf{g}_* in (37) with $\Delta \mathbf{u}$, whereby

$$\Delta \mathbf{u}^T \mathbf{g}_* = \Delta G . \quad (38)$$

In contrast to commonly used finite derivative formats the present format using \mathbf{g}_q implies fourth-order accuracy. There is some freedom in the choice of the last factor $\bar{\mathbf{K}}\Delta\mathbf{u}$, and e.g. the internal force increment $\Delta\mathbf{g}$ could be used instead with a corresponding change of the denominator. Finite derivative or gradient formulations typically make use of a mean state $\bar{\mathbf{u}}$, essentially equivalent to using $\mathbf{g}(\bar{\mathbf{u}})$ instead of the present representation \mathbf{g}_q . The present form has the advantage of being fourth-order accurate and directly available in global form.

It is observed that if a potential load were included directly via the load contribution \mathbf{f} , its time integral \mathbf{f}_* would be represented to fourth order by an expression similar to (31). The $\Delta\mathbf{v}$ contribution contained in the time integral would be cancelled by the extra $\Delta\mathbf{v}$ contribution already present in the energy balance equation (35), leaving a fourth-order approximation of the potential force similar to (33). Naturally, this fourth-order representation could be replaced by a finite derivative like that used for the internal force. Thus, the time integral representation of the force via \mathbf{f}_* supplemented by the increment $\Delta\mathbf{f}$ in the energy balance equation is a natural part of the fourth-order format.

5. Time stepping algorithm

The nonlinear state-space equations (30) are solved by Newton-Raphson iteration. The first step is to write the equations in the form of residual state-space vectors

$$\begin{bmatrix} \mathbf{r}_u \\ \mathbf{r}_v \end{bmatrix} = \begin{bmatrix} h\mathbf{f}_* \\ -\frac{1}{12}h^2\Delta\mathbf{f} \end{bmatrix} - \begin{bmatrix} \mathbf{C} & \mathbf{M} - \frac{1}{12}h^2\bar{\mathbf{K}} \\ \mathbf{M} - \frac{1}{12}h^2\bar{\mathbf{K}} & -\frac{1}{12}h^2\mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{v} \end{bmatrix} - h \begin{bmatrix} \mathbf{g}_*(\mathbf{u}) \\ -\mathbf{M}\bar{\mathbf{v}} \end{bmatrix}. \quad (39)$$

The idea of Newton-Raphson iteration is that the sum of the current value of the residual plus a linearized increment should vanish. This provides the following system of equations for the increments $\delta\mathbf{u}$ and $\delta\mathbf{v}$,

$$\begin{bmatrix} \mathbf{C} & \mathbf{M} - \frac{1}{12}h^2\mathbf{K} \\ \mathbf{M} - \frac{1}{12}h^2\mathbf{K} & -\frac{1}{12}h^2\mathbf{C} \end{bmatrix} \begin{bmatrix} \delta\mathbf{u} \\ \delta\mathbf{v} \end{bmatrix} + \frac{h}{2} \begin{bmatrix} 2\delta\mathbf{g}_*(\mathbf{u}) \\ -\mathbf{M}\delta\mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_u \\ \mathbf{r}_v \end{bmatrix}. \quad (40)$$

In this relation the current stiffness matrix \mathbf{K} in the lower left corner of the first matrix follows from the second order relation

$$\delta(\bar{\mathbf{K}}\Delta\mathbf{u}) \simeq \delta(\Delta\mathbf{g}) \simeq \mathbf{K}\delta\mathbf{u}, \quad (41)$$

and the upper off-diagonal block then follows from symmetry of the tangent stiffness relation.

The variational increment of the internal algorithmic force \mathbf{g}_* is expressed in terms of an equivalent algorithmic stiffness matrix \mathbf{K}_* ,

$$2\delta\mathbf{g}_*(\mathbf{u}) = \mathbf{K}_*\delta\mathbf{u}. \quad (42)$$

A full evaluation of \mathbf{K}_* based on the secant representation (37) would be rather elaborate, and an approximate form using the quartic-potential representation (33a) is used instead. When using this form the increment is found as

$$\delta(2\mathbf{g}_*) \simeq \frac{1}{3}\mathbf{K}\delta\mathbf{u} + \frac{4}{3}\mathbf{K}(\bar{\mathbf{u}})\delta\bar{\mathbf{u}} = \mathbf{K}\delta\mathbf{u} - \frac{2}{3}(\mathbf{K} - \mathbf{K}(\bar{\mathbf{u}}))\delta\mathbf{u}, \quad (43)$$

where \mathbf{K} is the current stiffness matrix. Within the quartic potential assumption the stiffness matrix $\mathbf{K}(\mathbf{u})$ is a quadratic function of the displacement \mathbf{u} . In the evaluation of the algorithmic internal force \mathbf{g}_* only the increment $\Delta\mathbf{K}$ enters, as illustrated in (33b), suggesting the linear approximation $\mathbf{K} - \mathbf{K}(\bar{\mathbf{u}}) \simeq \frac{1}{2}\Delta\mathbf{K}$. This corresponds to representing the algorithmic stiffness matrix by

$$\mathbf{K}_* \simeq \mathbf{K} - \frac{1}{3}\Delta\mathbf{K}. \quad (44)$$

This form of the algorithmic stiffness corresponds to a weighted average of the previous and the current value, $\mathbf{K}_* \simeq \frac{2}{3}\mathbf{K}_{n+1} + \frac{1}{3}\mathbf{K}_n$.

The solution of the incremental state-space equations (40) starts by using the second equation to express the velocity increment,

$$\delta\mathbf{v} = \frac{2}{h}(\mathbf{M} + \frac{1}{6}h\mathbf{C})^{-1} \left[(\mathbf{M} - \frac{1}{12}h^2\mathbf{K})\delta\mathbf{u} - \mathbf{r}_v \right]. \quad (45)$$

Substitution of this expression into the first of the incremental state-space equations (40) then gives the following linearized equation for the displacement increment $\delta\mathbf{u}$,

$$\begin{aligned} \left[\mathbf{K}_* + \frac{2}{h}\mathbf{C} + \left(\frac{2}{h}\right)^2 (\mathbf{M} - \frac{1}{12}h^2\mathbf{K})(\mathbf{M} + \frac{1}{6}h\mathbf{C})^{-1}(\mathbf{M} - \frac{1}{12}h^2\mathbf{K}) \right] \delta\mathbf{u} \\ = \frac{2}{h}\mathbf{r}_u + \left(\frac{2}{h}\right)^2 (\mathbf{M} - \frac{1}{12}h^2\mathbf{K})(\mathbf{M} + \frac{1}{6}h\mathbf{C})^{-1}\mathbf{r}_v. \end{aligned} \quad (46)$$

The fourth-order characteristics of the algorithm are obtained by the fourth-order representation of the load (33), and by the following two modified forms of the mass matrix,

$$\mathbf{M}_c = \mathbf{M} + \frac{1}{6}h\mathbf{C}, \quad \mathbf{M}_k = \mathbf{M} - \frac{1}{12}h^2\mathbf{K}. \quad (47)$$

Table 1: Fourth-order conservative algorithm.

1)	Initial conditions: $\mathbf{u}_0, \mathbf{v}_0$.
2)	Prediction step: $\Delta \mathbf{u} = h \mathbf{v}_n, \Delta \mathbf{v} = \mathbf{0}$.
3)	Residual calculation: $\mathbf{u} = \Delta \mathbf{u} + \mathbf{u}_n$. $\mathbf{g}_q = \bar{\mathbf{g}} - \frac{1}{12} \Delta \mathbf{K} \Delta \mathbf{u}$. If $G(\mathbf{u})$ higher order than quartic and $ \Delta \mathbf{u}^T \bar{\mathbf{K}} \Delta \mathbf{u} > \varepsilon_\eta$, $\eta = [\Delta G - \Delta \mathbf{u}^T \mathbf{g}_q] / (\Delta \mathbf{u}^T \bar{\mathbf{K}} \Delta \mathbf{u})$ else $\eta = 0$. $\mathbf{r}_u = h \mathbf{f}_* - h \mathbf{g}_q - (\eta h \bar{\mathbf{K}} + \mathbf{C}) \Delta \mathbf{u} - \bar{\mathbf{M}}_K \Delta \mathbf{v}$. $\mathbf{r}_v = -\frac{1}{12} h^2 \Delta \mathbf{f} - \bar{\mathbf{M}}_K \Delta \mathbf{u} + \frac{1}{2} h \mathbf{M}_C \Delta \mathbf{v} + h \mathbf{M} \mathbf{v}_n$.
4)	Displacement and velocity sub-increments: $\mathbf{K}_* = \mathbf{K} - \frac{1}{3} \Delta \mathbf{K}$. $\mathbf{K}_u = [\mathbf{K}_* + (2/h) \mathbf{C} + (2/h)^2 \mathbf{M}_K \mathbf{M}_C^{-1} \mathbf{M}_K]$. $\delta \mathbf{u} = \mathbf{K}_u^{-1} [(2/h) \mathbf{r}_u + (2/h)^2 \mathbf{M}_K \mathbf{M}_C^{-1} \mathbf{r}_v]$. $\delta \mathbf{v} = (2/h) \mathbf{M}_C^{-1} [\mathbf{M}_K \delta \mathbf{u} - \mathbf{r}_v]$. $\Delta \mathbf{u} = \Delta \mathbf{u} + \delta \mathbf{u}, \Delta \mathbf{v} = \Delta \mathbf{v} + \delta \mathbf{v}$. If $ \mathbf{r}_u^T \mathbf{r}_u + \mathbf{r}_v^T \mathbf{r}_v > 2\varepsilon_r^2$ or $ \delta \mathbf{u}^T \delta \mathbf{u} + \delta \mathbf{v}^T \delta \mathbf{v} > 2\varepsilon_k^2$ repeat from 3).
5)	State vector update: $\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta \mathbf{u}, \mathbf{v}_{n+1} = \mathbf{v}_n + \Delta \mathbf{v}$.
6)	Return to 2) for new time step, or stop.

The time integration algorithm is summarized in pseudo-code format in Table 1, using the fourth-order load representation \mathbf{f}_* from (33) and the modified mass matrices \mathbf{M}_C and \mathbf{M}_K from (47).

After setting the initial conditions in (1) the time step starts in (2) with a simple prediction of the displacement increment from the known velocity \mathbf{v}_n . The residuals \mathbf{r}_u

and \mathbf{r}_v of the state-space equation (39) are calculated in (3) using the quartic potential internal force \mathbf{g}_q with a conditional extension to general secant form. The displacement and velocity sub-increments $\delta\mathbf{u}$ and $\delta\mathbf{v}$ are then calculated in (4) by using (46) and (45), respectively. If the convergence tolerance, set by the residual threshold ε_r and the kinematic threshold ε_k are not met, a new iteration starts from (3), otherwise the load step is concluded with an update of the displacement and velocity vectors in (5).

Remark

A characteristic of the present algorithm is that it corresponds to a cubic Hermite representation of the displacement history $\mathbf{u}(t)$ within each time increment $[t_n, t_{n+1}]$ in terms of the end-point displacements $[\mathbf{u}_n, \mathbf{u}_{n+1}]$ and the end-point velocities $[\mathbf{v}_n, \mathbf{v}_{n+1}]$. The cubic representation of the displacement may be expressed in terms of the normalized time variable $\tilde{\tau} = (t - \bar{t})/h$ as

$$\mathbf{u}(\tilde{\tau}) = (\bar{\mathbf{u}} - \frac{1}{8}h\Delta\mathbf{v}) + \frac{1}{2}\tilde{\tau}(3\Delta\mathbf{u} - h\bar{\mathbf{v}}) + \frac{1}{2}\tilde{\tau}^2h\Delta\mathbf{v} - 2\tilde{\tau}^3(\Delta\mathbf{u} - h\bar{\mathbf{v}}), \quad -\frac{1}{2} \leq \tilde{\tau} \leq \frac{1}{2} \quad (48)$$

It is observed that in the second-order form of the present algorithm (30), in which all terms with the factor $\frac{1}{12}h^2$ are omitted, the second state-space equation reduces to the kinematic central difference relation $\Delta\mathbf{u} = h\bar{\mathbf{v}}$, whereby the cubic term in the representation (48) vanishes identically. Thus, the typical second-order algorithms have at most parabolic interpolation within the integration intervals.

6. Examples

The properties of the energy-momentum algorithm developed above are illustrated by three simple dynamic systems in the following examples. The first two examples illustrate the fourth-order accuracy and energy conservation of the algorithm by application to the Duffing oscillator having a quartic energy function and an oscillator with tanh-stiffness with an explicit non-polynomial energy function. The fourth-order convergence is demonstrated by calculation of the relative period error $\Delta T/T$ in terms of the normalized time step. The period is determined from the intersection of the displacement time history with the first axis using the cubic representation (48).

The last example is concerned with the calculation of high-frequency linear molecular vibrations taking place in a system with weak highly non-linear forces between the molecules. The example illustrates the superior representation of the correct time scale obtained by the fourth-order algorithm.

6.1. Duffing oscillator

The Duffing oscillator is often used to model the dynamics of a single-degree-of-freedom system with a simple non-linear stiffness represented as a combination of a linear and a cubic term. It is represented by the equation of motion for the mass m as

$$m \frac{d^2 u}{dt^2} + g(u) = f(t), \quad (49)$$

with internal force $g(u)$ and internal energy $G(u)$ given in terms of the initial stiffness k and the stiffness non-linearity parameter λ as

$$g(u) = ku(1 + \lambda^2 u^2), \quad G(u) = \frac{1}{2}ku^2(1 + \frac{1}{2}\lambda^2 u^2). \quad (50)$$

The tangent stiffness follows from differentiation of the internal force $g(x)$ as

$$K(u) = k(1 + 3\lambda^2 u^2). \quad (51)$$

The energy potential $G(u)$ is of degree four in the displacement u , and it is easily verified that the secant correction vanishes.

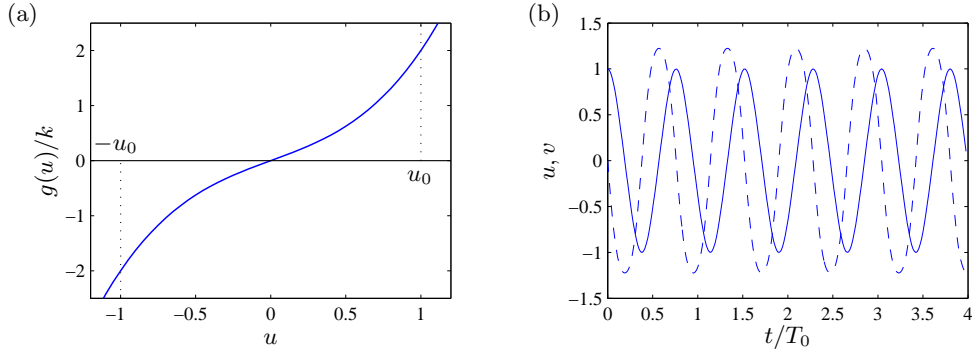


Figure 1: Duffing oscillator, $\lambda = 1.0$. a) Internal force $g(u)$; b) Free response $u(t)$ —, $v(t)/\omega_0$ - -.

Figure 1 illustrates the Duffing oscillator with $m = 1.0$, $k = 1.0$, $\lambda = 1.0$. This corresponds to a reference angular frequency $\omega_0 = 1.0$, and the reference period $T_0 = 2\pi$.

The non-linear stiffness relation is shown in Fig. 1(a) and the free response is shown in Fig. 1(b) for initial conditions $u_0 = 1.0$, $v_0 = 0.0$. The free-vibration period T of the Duffing oscillator depends on the energy level and can be expressed explicitly in terms of the complete elliptic integral by variable substitutions in the energy integral,

$$T = \frac{4}{\omega} K_e(m_e) \quad (52)$$

in which $K_e(m_e)$ is the complete elliptic integral [24]. The parameters are conveniently expressed in terms of the small-amplitude angular frequency $\omega_0 = \sqrt{k/m}$ and the normalized maximum amplitude λu_m as

$$\omega = \omega_0 \sqrt{1 + (\lambda u_m)^2}, \quad m_e = \frac{(\lambda u_m)^2}{2(1 + (\lambda u_m)^2)}. \quad (53)$$

With the present displacement initial condition the maximum amplitude is $u_m = u_0$. At the normalized amplitude level $\lambda u_0 = 1$ the period determined by the elliptic integral is $T = 0.7589 T_0 = 4.768$.

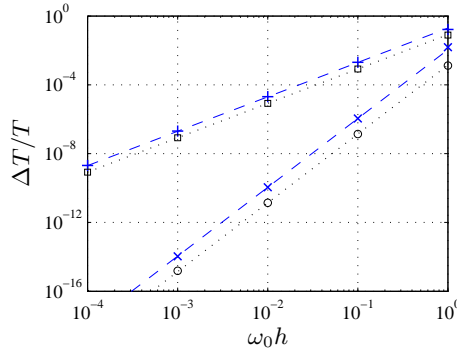


Figure 2: Duffing oscillator $\lambda u_m = 1.0$: 4th order (\times), 2nd order ($+$). Linear oscillator: 4th order (o), 2nd order (\square).

The relative error on the numerically determined period T_n defined as $\Delta T/T = (T_n - T)/T$ is calculated for the time step lengths $\omega_0 h = [10^{-3}, 10^{-2}, 10^{-1}, 10^0]$ and is shown with the \times symbol in Fig. 2. The time histories were obtained using $\varepsilon_r = \varepsilon_k = 10^{-12}$, leading to 7 iterations for $\omega_0 h = 0.5$ and 4 iterations for $\omega_0 h = 0.1$. It is important to note that in order to retain the 4th order accuracy when using the 4th order algorithm, the response between integration points must be determined by cubic Hermite

interpolation, as defined by the interval end-point values $[u_n, v_n]$ and $[u_{n+1}, v_{n+1}]$ in (48). The calculated points in the graph fall on a line corresponding to the relation $\Delta T/T = 0.0111(\omega_0 h)^4$, confirming the 4th order property of the algorithm. When leaving out the terms with the factor $\frac{1}{12}h^2$ in (39), but keeping the full representation of the internal force \mathbf{g}_* , a conservative 2nd order time integration algorithm is obtained, [15]. The results from this reduced form are shown in Fig. 2 with a + symbol. The approximation involved in the derivation of the second-order algorithm does not go beyond linear interpolation of the response u within each integration interval, and the kinematic relation in the second state-space equation corresponds to quadratic interpolation, as the coefficient of the cubic term vanishes. The period error from the second-order algorithm falls on a line corresponding to the relation $\Delta T/T = 0.204(\omega_0 h)^2$, confirming the 2nd order accuracy of the reduced form of the algorithm. An important observation following from these results is that for $\omega_0 h = 1.0$, corresponding closely to the coarsest useful discretization with about 5 points per period, the period error of the 4th order algorithm is about 1 % as compared to a period error of about 20 % for the 2nd order algorithm. Furthermore, the ratio of the two errors increases with the square of the time increment used, making the period accuracy of the 4th order algorithm remarkably better than the corresponding 2nd order method. Both the 4th order and the 2nd order algorithms are energy conserving to within the computational accuracy of the MATLAB program with $\Delta E/E \simeq 10^{-14}$.

For linear oscillations, the asymptotic period error is known explicitly for the 2nd order mean value algorithm [25], as well as for the linear form of the present 4th order algorithm [18],

$$\frac{\Delta T}{T} = \frac{T_n - T}{T} \sim \begin{cases} \frac{1}{12}(\omega_0 h)^2, \\ \frac{1}{720}(\omega_0 h)^4. \end{cases} \quad (54)$$

These relations are shown in Fig 2 as dotted lines, together with numerical results from the 2nd order and the 4th order algorithms. It is seen that the asymptotic results are very accurate all the way up to $\omega_0 h = 1$, corresponding to about 6 integration points per period. It is to be noted that while the nonlinear algorithms are of the same order of accuracy as their linear counterparts, the numerical constants are somewhat larger

for the nonlinear equation, partly due to the fact that the equivalent angular frequency of the nonlinear oscillation is larger than the reference value ω_0 .

6.2. Oscillator with tanh stiffness

The present example explores the algorithm in Table 1 in connection with an oscillator with the internal force represented by a tanh-function. The equation of motion has the generic form (49). In the present case the internal force and the internal energy are given in terms of the initial stiffness k and the stiffness non-linearity parameter λ as

$$g(u) = \frac{k}{\lambda} \tanh(\lambda u), \quad G(u) = \frac{k}{\lambda^2} \ln(\cosh(\lambda u)). \quad (55)$$

The tangent stiffness follows from differentiation of the internal force $g(x)$ as

$$K(u) = \frac{k}{\cosh^2(\lambda u)}. \quad (56)$$

The internal force $g(\lambda u)$ is shown as a function of the displacement u in Fig. 3(a) for the parameter $\lambda = 4$ and with $u_0 = 1$ indicating the initial value used in the example. It is seen that the system is severely nonlinear with nearly constant internal force for $|u| \gtrsim 0.5u_0$.

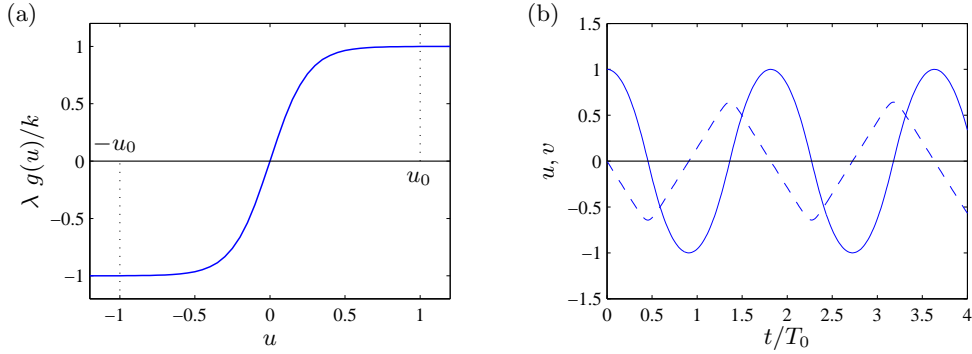


Figure 3: Tanh oscillator, $\lambda = 4.0$. a) Internal force $g(u)$; b) Free response $u(t)$ —, $v(t)/\omega_0$ - -.

Figure 3b shows time histories of the displacement and velocity of the free undamped response of the tanh-oscillator with $\lambda = 4$ as shown in Fig. 3(a). For $\lambda \simeq 0$ the oscillator is linear with angular frequency $\omega_0 = 2\pi/T_0 = (k/m)^{1/2}$. For increasing λ the period increases and in the present case of $\lambda = 4$ the period is nearly doubled with $T = 1.818 T_0$.

As seen from the time histories in Fig. 3(b) the velocity is no longer just a phase shift of the displacement history but develops a saw-tooth shape with smaller maximum amplitude than in the similar linear problem.

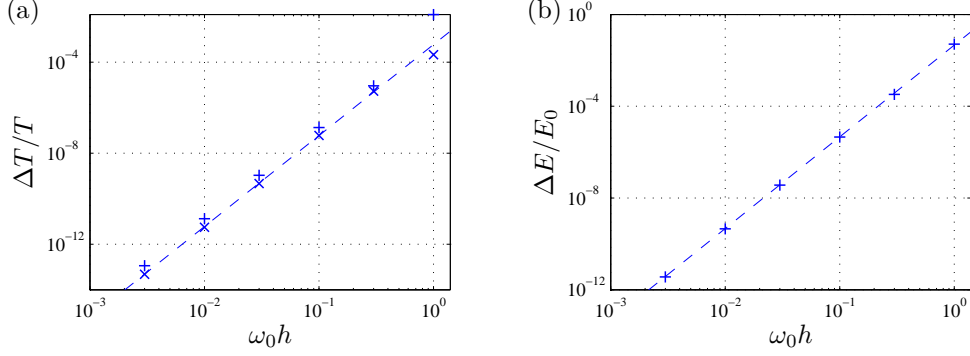


Figure 4: Tanh-oscillator, $\lambda u_m = 4.0$. a) Period error $\Delta T/T$: conservative (\times), without secant term ($+$); b) Energy error $\Delta E/E$ without secant term.

The period error $\Delta T/T = (T_n - T)/T$ is illustrated in Fig. 4(a) using the time step lengths $\omega_0 h = [3 \cdot 10^{-3}, 10^{-2}, 3 \cdot 10^{-3}, 10^{-1}, 3 \cdot 10^{-1}, 10^0]$. The time histories were obtained using $\varepsilon_r = \varepsilon_k = 10^{-12}$, leading to 5 iterations for $\omega_0 h = 0.5$ and 3 iterations for $\omega_0 h = 0.1$. The results for the fully conservative form, including the secant stiffness in the representation of the internal force \mathbf{g}_* given in (37), are shown by the \times symbol, while the results from the fourth-order formulation in terms of the internal force \mathbf{g}_q without secant term are shown by the $+$ symbol. The relative period error of the conservative algorithm is well represented by the relation $\Delta T/T = 5.8 \cdot 10^{-4}(\omega_0 h)^4$ shown in the figure. A similar relation holds for the form without the secant term, but with the coefficient $1.3 \cdot 10^{-3}$. In the full version of the algorithm, including the secant term, the energy is conserved to within a relative error of $\Delta E/E \sim 10^{-15}$, corresponding to the machine accuracy. The relative energy error for the case in which the secant term is omitted is shown in Fig. 4(b). The relative energy error is well represented by the asymptotic relation $\Delta E/E = 0.045(\omega_0 h)^4$, confirming the fourth-order accuracy.

6.3. A Fermi-Pasta-Ulam problem

This example concerns oscillations in a so-called modified Fermi-Pasta-Ulam system consisting of point masses connected by alternating stiff linear and soft non-linear springs. The problem has been used to characterize various time-integration procedures in [26, 14]. The system is illustrated in Fig. 5. It consists of $2n$ identical point masses m , described by the displacement vector $\mathbf{u} = [u_1, \dots, u_{2n}]^T$ and the corresponding velocity vector $\mathbf{v} = [v_1, \dots, v_{2n}]^T$. The masses are pairwise connected by linear springs with stiffness k and non-linear springs with cubic spring stiffness parameter κ . The corresponding total energy is

$$E = \frac{m}{2} \sum_{i=1}^{2n} v_i^2 + \frac{k}{2} \sum_{j=1}^n (u_{2j} - u_{2j-1})^2 + \kappa \sum_{j=0}^n (u_{2j+1} - u_{2j})^4 \quad (57)$$

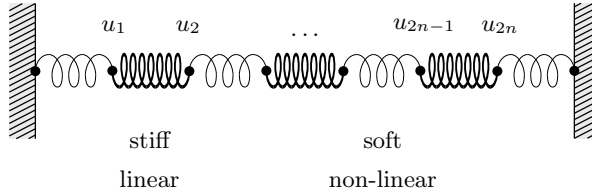


Figure 5: Chain of alternating soft nonlinear and stiff linear springs.

The parameters of the problem are chosen such that the vibrations consist of rapid extension/compression oscillations over the linear springs, while the motion of their centers move slowly as controlled by the soft non-linear springs. Following [14] the problem is transformed to reflect this combination of fast and slow motion by introducing the generalized displacements and velocities

$$\begin{aligned} q_i &= (u_{2i} - u_{2i-1})/\sqrt{2}, & q_{i+3} &= (u_{2i} + u_{2i-1})/\sqrt{2}, \\ p_i &= (v_{2i} - v_{2i-1})/\sqrt{2}, & p_{i+3} &= (v_{2i} + v_{2i-1})/\sqrt{2}, \end{aligned} \quad (58)$$

for $i = 1, \dots, n$. By this transformation the fast oscillatory vibrations of the linear springs are given by the first three components of the generalized displacements, while the background motion is contained in the last three components. In terms of these

variables the energy takes the form

$$E = \frac{m}{2} \sum_{i=1}^{2n} p_i^2 + k \sum_{j=1}^n q_j^2 + \frac{\kappa}{4} \left\{ (q_1 - q_4)^4 + (q_1 + q_2 + q_4 - q_5)^4 + (q_2 + q_3 + q_5 - q_6)^4 + (q_3 + q_6)^4 \right\}. \quad (59)$$

The problem considered in this example corresponds to [14] with $n = 3$ linear springs and parameters $m = 1$, $\kappa = 1$ and $k = \omega^2/2$ with $\omega = 50$. For displacements of order unity this corresponds to rapid oscillations in the linear springs with angular frequency close to ω , while the centers of the linear springs move at a much slower time scale.

In the terminology of the present paper the mass matrix is the 6×6 unit matrix, $\mathbf{M} = \mathbf{I}$. The internal energy consists of a quadratic and a quartic contribution. The internal force and stiffness matrix from the quadratic energy are

$$\mathbf{g}_2 = [q_1, q_2, q_3, 0, 0, 0]^T, \quad \mathbf{K}_2 = \omega^2 [1, 1, 1, 0, 0, 0]. \quad (60)$$

The spring forces \mathbf{g}_4 from the non-linear springs and the associated stiffness matrix \mathbf{K}_4 are conveniently expressed in terms of the shortening of the springs, collected in the 4-component vector

$$\boldsymbol{\gamma} = \mathbf{A}\mathbf{q}, \quad \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 1 & 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (61)$$

It follows from differentiation of the quartic part of the energy that the corresponding internal force and stiffness matrix take the form

$$\mathbf{g}_4 = \mathbf{A}^T [\gamma_1^3, \gamma_2^3, \gamma_3^3, \gamma_4^3]^T, \quad \mathbf{K}_4 = 3\mathbf{A}^T [\gamma_1^2, \gamma_2^2, \gamma_3^2, \gamma_4^2] \mathbf{A}. \quad (62)$$

The total internal force then is $\mathbf{g} = \mathbf{g}_2 + \mathbf{g}_4$, and the tangent stiffness matrix is similarly given as $\mathbf{K} = \mathbf{K}_2 + \mathbf{K}_4$. This completes the mathematical formulation of the problem which is now solved by the fourth-order conservative algorithm in Table 1 and some second-order alternatives.

The specific numerical example is a free vibration problem used in [14] to illustrate the performance of several second-order time-integration algorithms. The system is

started by introducing the following non-zero initial conditions

$$q_1(0) = 1/\omega, \quad p_1(0) = 1, \quad q_4(0) = 1, \quad p_4(0) = 1, \quad (63)$$

imposing an extension of the first linear spring and initial velocities of its end-points. The *local* energy associated with each of the stiff linear springs is

$$E_j = \frac{1}{2}p_j^2 + \frac{1}{2}\omega^2 q_j^2, \quad j = 1, 2, 3. \quad (64)$$

The initial conditions (63) correspond to $E_1 = 1$ and $E_2 = E_3 = 0$, which is approximately half of the total energy $E = 2 + 3\omega^{-2} + \frac{1}{2}\omega^{-4} = 2.0012$. For the present choice of parameters the oscillations of the linear springs are only weakly coupled through the nonlinear springs, and the oscillations of the linear springs then occur at an angular frequency close to ω , corresponding to a period of $T \simeq 0.126$.

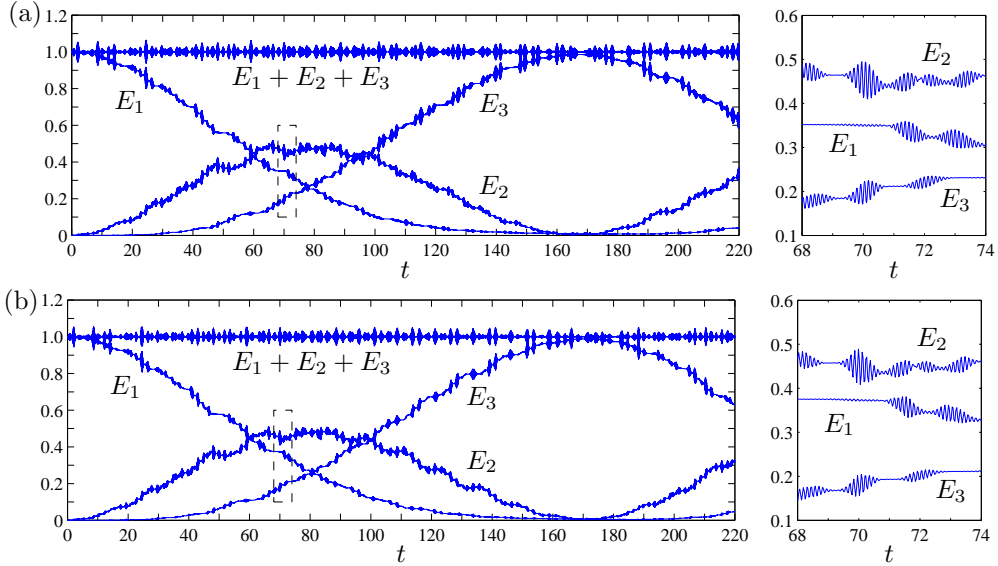


Figure 6: Fermi-Pasta-Ulam system, 4th order conservative: a) $h = 0.01$, b) $h = 0.03$.

Figures 6(a,b) show the energy of each of the linear springs as well as their sum, calculated by the fourth-order conservative algorithm of Table 1 with time increments $h = 0.01$ and $h = 0.03$, respectively. The time increment $h = 0.01$ corresponds to about 12 integration points per period of the linear oscillators, and a comparison with computations using the very small time increment $h = 0.001$, used in [14] in connection with

second-order algorithms to establish a reference case, do not show any difference from the results in Fig. 6(a). The sub-figures to the right show clumps of rapid fluctuations in the energy level in each of the linear springs, occurring with an oscillation period of 0.125, corresponding closely to the theoretical period of free vibrations of the linear springs. For the time increment $h = 0.01$ the estimated period error from (54b) is about 10^{-4} for the present fourth-order algorithm. Figure 6(b) shows the similar results for the time increment $h = 0.03$, corresponding to about 4 points per oscillation period. A small change can be observed in the global time scale – e.g. around the minimum of the E_2 -curve at $t \simeq 170$ and in the level of the energy-curves in the sub-figures to the right – but the general agreement is very good, even at this rather coarse discretization. The reason is most likely to be found in the fully consistent cubic Hermite interpolation within each time increment as discussed in connection with the interpolation representation (48). In both computations the convergence threshold was set to 10^{-14} , leading to a similar accuracy on the energy conservation.

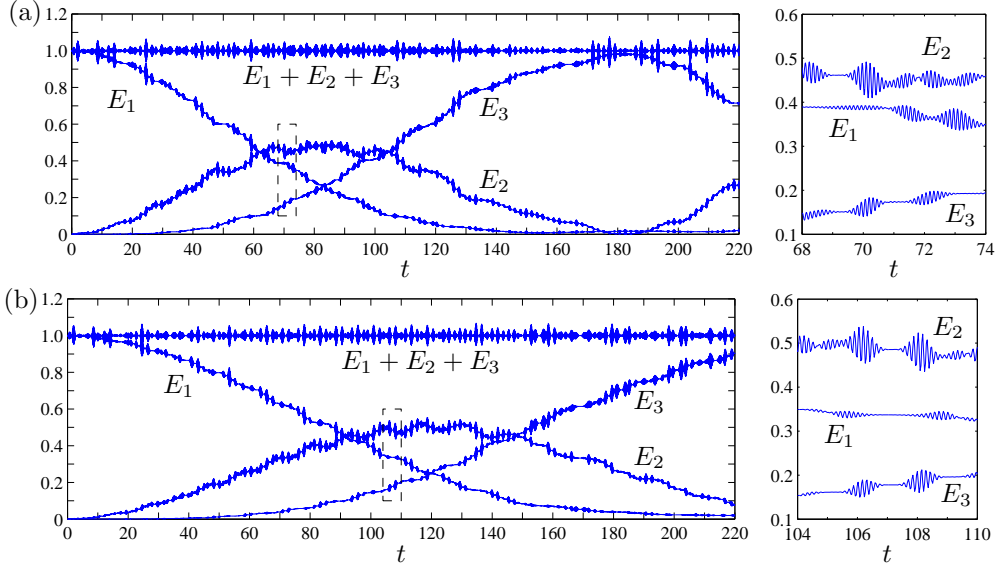


Figure 7: Fermi-Pasta-Ulam system, 2nd order midpoint: a) $h = 0.01$, b) $h = 0.03$.

The computations were repeated with the second-order mid-point algorithm, in which the internal force is represented as $\mathbf{g}(\bar{\mathbf{u}})$. This algorithm was selected as the best per-

forming of the three immediate alternatives: mid-point, trapezoidal and the energy conserving form using the combination $\frac{1}{3}\bar{\mathbf{g}} + \frac{2}{3}\mathbf{g}(\bar{\mathbf{u}})$. Figure 7 shows the results for $h = 0.01$. It is observed that the global time scale is stretched such that e.g. the intersection between the E_1 and the E_3 curves moves from $t = 78$ to $t = 83$, and similarly the minimum of the E_2 curve moves from $t \simeq 171$ to $t \simeq 182$, both corresponding to a stretch of the time scale with about 6.4 %. This is larger than the period elongation of 2 % predicted for the linear oscillations by (54a). The local energy fluctuation pattern shown in the right side of Fig. 7(a) exhibits the same qualitative oscillations as in the reference solution in Fig. 6(a), but the basic levels have changed, and small differences in the oscillation patterns can be seen. The results from the mid-point algorithm with integration time increment $h = 0.03$ are shown in Fig. 7(b). For this time increment the solution has deteriorated considerably. The global time scale appears to have increased by about 50 %, and the local energy fluctuation pattern, shown to the right for a window moved accordingly, have changed character, as the difference in local and global scaling has increased. It seems clear from the results that the global time scale is more sensitive to the integration time increment than the time scale of the rapid oscillations. The mid-point method is not energy conserving, but the fluctuations in total energy of $\Delta E/E = 0.80 \cdot 10^{-4}$ and $\Delta E/E = 0.54 \cdot 10^{-3}$, respectively, are much smaller than the error in the global time scale.

7. Conclusion

Most conservative time integration algorithms for linear as well as for nonlinear dynamic systems are of second order accuracy. This leaves a trade-off between the use of explicit methods that typically require small time step relative to the smallest natural vibration period in the system, and implicit methods where a larger time step may not lead to instability but typically to a noticeable phase error. While the implicit methods can be made energy conserving, and thereby unconditionally stable, by representing the internal force terms via a discrete gradient, this generally does not reduce the phase error that builds up over time. The present paper develops a fourth-order time integration method for dynamic systems with constant mass and damping matrices – a format often

encountered in mathematical physics and engineering. The method can be extended to more general dynamic systems, by using a formulation in terms of Hamilton's equations. The derivation makes use of integration by parts of the state-space equations of motion. This process introduces some new terms relative to those of second-order algorithms. The most important corrections introduced in this formulation are a modification of the representation of the inertial effects and a fourth-order representation of the time-integral of the internal force.

In the fourth-order algorithm the mass matrix in the coupling terms between the two state-space equations is replaced by a modified matrix, in which the classic mass matrix is reduced by a term that is quadratic in the integration time increment. This modification dramatically reduces the elongation of the oscillation periods typically generated by implicit second-order algorithms. This improvement is closely linked to the fact that in the fourth-order algorithm the representation of the dynamic displacement paths corresponds to cubic Hermite interpolation, with displacements and velocities being continuous between the integration intervals. In contrast, second-order methods typically use a central difference definition of the mean velocity, corresponding to a parabolic representation of the displacement path. The consistent cubic Hermite representation associated with the fourth-order algorithm enables accurate representation of oscillations with as little as 3-4 points per oscillation period.

The other key aspect of the present method is the representation of the internal force. The derivation via integration by parts leads to several possible representations, using end-point values or mid-point average displacement together with a term containing the increment of the stiffness matrix. These various forms are all fourth-order accurate, and for a force potential that is quartic in the displacement components this leads to energy conservation properties of the algorithm. For more general forms of the force potential, energy conservation is established by a higher-order secant based correction term. The format is reminiscent of the so-called discrete gradient, but is formulated on the basis of a fourth-order representation in order to retain the accuracy of the algorithm. As a consequence the present secant correction term is typically very small.

References

- [1] R.A. LaBudde, D. Greenspan, Discrete mechanics – a general treatment. *Journal of Computational Physics*, 77 (1974) 85–102.
- [2] R.A. LaBudde, D. Greenspan, Energy and momentum conserving methods of arbitrary order for the numerical integration of equations of motion. Parts I and II. *Numerische Mathematik*, 25 (1976) 323–346 and 26 (1976) 1–16.
- [3] J.C. Simo, K.K. Wong, Unconditionally stable algorithms for rigid body dynamics that exactly preserve energy and momentum. *International Journal for Numerical Methods in Engineering*, 31 (1991) 19–52.
- [4] J.C. Simo, N. Tarnow, The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift für angewandte Mathematik und Physik*, 43 (1992) 757–792.
- [5] J.C. Simo, N. Tarnow, K.K. Wong, Exact energy-momentum conserving algorithms and symplectic schemes for nonlinear dynamics. *Computer Methods in Applied Mechanics and Engineering*, 100 (1992) 63–116.
- [6] I. Romero, An analysis of the stress formula for energy-momentum methods in nonlinear elastodynamics. *Computational Mechanics*, 50 (2012) 603–610.
- [7] O. Gonzalez, Time integration and discrete Hamiltonian systems, *Journal of Non-linear Science*, 6 (1996) 449–467.
- [8] O. Gonzalez, Exact energy and momentum conserving algorithms for general models in non-linear elasticity, *Computer Methods in Applied Mechanics and Engineering*, 190 (2000) 1763–1783.
- [9] R.I. McLachlan, G.R.W. Quispel, N. Robidoux, Geometric integration using discrete gradients. *Philosophical Transactions of the Royal Society, London* 357 (1999) 1021–1045.
- [10] D. G. Luenberger, *Linear and Nonlinear Programming*. 2nd ed., Addison-Wesley, Reading, Massachusetts, 1989.

- [11] P. Betsch, P. Steinmann, Inherently energy conserving time finite elements for classical mechanics. *Journal of Computational Physics*, 160 (2000) 88–116.
- [12] P. Betsch, P. Steinmann, Conservation properties of time FE method – Part II: Time-stepping schemes for non-linear elastodynamics. *International Journal for Numerical Methods in Engineering*, 50 (2001) 1931–1955.
- [13] F. Armero, I. Romero, On the formulation of high-frequency dissipative time-stepping algorithms for non-linear dynamics. Part II: Second order methods. *Computer Methods in Applied Mechanics and Engineering*, 190 (2001) 6783–6824.
- [14] E. Hairer, C. Lubich, G. Wanner, *Geometric Numerical Integration*, 2nd ed., Springer, Berlin, 2006.
- [15] S. Krenk, Global format for energy-momentum based time integration in nonlinear dynamics. *International Journal for Numerical Methods in Engineering* 100 (2014) 458–476.
- [16] J. de Frutos, J.M. Sanz-Serna, An easily implementable fourth-order method for the time integration of wave problems. *Journal of Computational Physics* 103 (1992) 160–168.
- [17] N. Tarnow, J.C. Simo, How to render second order accurate tim-stepping algorithms fourth order accurate while retaining the stability and conservation properties. *Computer Methods in Applied Mechanics and Engineering*, 115 (1994) 233–252.
- [18] S. Krenk, State-space time integration with energy control and fourth-order accuracy for linear dynamic systems. *International Journal for Numerical Methods in Engineering*. 65 (2006) 595–619.
- [19] M. Groß, P. Betsch, P. Steinmann, Conservation properties of a time FE method. Part IV, Higher order energy and momentum conservation schemes. *International Journal for Numerical Methods in Engineering*, 63 (2005) 1849–1897.
- [20] A. Depouhon, E. Detournay, V. Denoël, Accuracy of one-step integration schemes for damped/forced linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 99 (2014) 33–353.

- [21] C. Bottasso, L. Trainelli, An attempt at the classification of energy decaying schemes for structural and multibody dynamics. *Multibody System Dynamics*, 12 (2004) 173–185.
- [22] E.V. Lens, A. Cardona, M. Géradin, Energy preserving time integration for constrained multibody systems. *Multibody System Dynamics*, 11 (2004) 41–64.
- [23] S. Krenk, Global format for conservative time integration in nonlinear dynamics, *Proceedings of the 9th International Conference on Structural Dynamics, EURO-DYN 2014*, Porto, Portugal, 30 June – 2 July, 2014. A. Cunha, E. Caetano, P. Ribeiro, G. Müller (eds.). ISSN: 2311-9020; ISBN: 978-972-752-165-4.
- [24] M. Abramowitz, I. Stegun, *Handbook of Mathematical Functions*, Dover, New York, 1989.
- [25] M. Géradin, D. Rixen, *Mechanical Vibrations*, 2nd ed., Wiley, Chichester, UK, 1997.
- [26] E. Hairer, C. Lubich, Long-time energy conservation of numerical methods for oscillatory differential equations. *SIAM Journal of Numerical Analysis* 38 (2000) 414–441.